

Modèle relationnel

Le **modèle relationnel** est une manière de modéliser les relations existantes entre plusieurs informations, et de les ordonner entre elles. Cette modélisation qui repose sur des principes mathématiques mis en avant par E.F. Codd est souvent retranscrite physiquement («implémentée») dans une base de données

Sommaire

Brève description

Opérateurs relationnels

Règles de gestion d'un système relationnel de base des données

Principe du modèle relationnel

La modélisation relationnelle et sa transcription en base de données

Terminologie structurelle^[5]

Domaine

Tuple

Attributs

Clé candidate

Relation

Degré

Cardinalité de relation

Relation 1:1

Relation 1:N

Relation N:N

Notes et références

Voir aussi

Liens externes

Brève description

On appelle « relation » un ensemble d'attributs qui caractérisent une proposition ou une combinaison de propositions comme "un employé a un matricule, il a un nom, il a un employeur". Dans cet exemple, les attributs de l'employé sont : son matricule, son nom et son employeur. Chaque combinaison de propositions ainsi formée est appelée tuple ou collection ordonnée d'objets. Par exemple l'ensemble ("1245", "Jean Dupond", "Compagnie des belles lettres") constitue un tuple de relation "employé". Les relations sont d'ordinaire représentées sous la forme de tables. Dans l'exemple précédent, la table serait libellée "employé". Usuellement, les praticiens accordent la même signification aux concepts de "relation" et de "table". De même, ils assimilent d'une part la "ligne dans la table" et le tuple, et d'autre part le "libellé de colonne dans la table" et l'attribut. Par définition, chaque tuple d'une relation est unique. Il est identifié par un attribut (un identifiant unique appelé "clef primaire") ou une combinaison de plusieurs attributs qui forme la clef. L'ordre des tuples n'est pas significatif.

Codd a défini une algèbre relationnelle et des opérateurs qui permettent de construire de nouvelles relations en combinant des relations préalablement définies. Les idées de Codd ont été implémentées -- plus ou moins fidèlement -- dans les systèmes de gestion des bases de données relationnelles ou SGBDR telles que le projet expérimental IBM System R, puis des produits commerciaux tels

qu'Oracle, DB2 ou MySQL, et dans le langage de manipulation des données SQL.

Le modèle relationnel est aujourd'hui l'un des modèles les plus utilisés. « Les premiers systèmes de gestion de base de données (SGBD ou DBMS en anglais) bâtis sur ce modèle ont été SQL/DS et DB2 de IBM, d'où est né le langage de manipulation de bases relationnelles, SQL (Structured Query Language). »¹ Le modèle relationnel est basé sur deux instruments puissants : l'algèbre relationnelle (c'est-à-dire le concept mathématique de relation en théorie des ensembles) et la notion de produit cartésien. Ce modèle définit une façon de représenter les données, les opérations qui peuvent être effectuées ainsi que les mécanismes pour préserver la consistance des données. E.F Codd a décrit les principes et la conception de modèle relationnel dans son livre « A relational model of data for large shared data banks ».

Opérateurs relationnels

Selon E. F. Codd, une base de données définie selon le modèle relationnel est manipulée à l'aide d'un ensemble d'opérations formelles sur les relations. Les opérations relationnelles permettent par exemple de créer une nouvelle relation, qui est représentée dans ce modèle, par une table à deux dimensions².

Le modèle relationnel définit 5 opérateurs de base qui sont l'union, la différence, la sélection (ou restriction), la projection et le produit cartésien. Ces opérateurs présentent l'avantage d'être fermés (ils agissent sur les relations et renvoient des relations qui peuvent de nouveau être combinées grâce aux opérateurs).

L'opérateur d'union permet de combiner deux relations (ou "tables") de même schéma (c'est-à-dire deux relations comportant les mêmes attributs) et renvoie une relation de même schéma que les relations initiales et dont l'ensemble des n-uplets (c'est-à-dire tuples ou "lignes du tableau") est l'union ensembliste des n-uplets des relations qui ont été combinées (ensembliste signifie qu'il n'y a pas d'ordre dans la présentation des lignes du tableau et qu'il ne peut pas y avoir de doublons).

L'opérateur de différence est similaire (mais non symétrique) et retourne les n-uplets qui figurent dans la première relation mais pas dans la seconde.

L'opérateur de sélection (unaire) permet de retourner un sous-ensemble des n-uplets de la relation initiale. Ils doivent vérifier un critère construit sur la base d'une conjonction (et), d'une disjonction (ou), d'une négation de triplets (attribut, comparateur, valeur). Le comparateur peut être >, <, =, <>, ... La notion de valeur peut être soit une constante de type valeur numérique, chaîne de caractères, ... soit un attribut (mais alors la comparaison a lieu sur le même n-uplet). Il y a alors diminution du nombre de lignes mais le nombre de colonnes reste identique.

L'opérateur de projection (unaire) permet de réduire le nombre d'attributs retenus dans la sélection. Le nombre de colonnes est alors réduit. Cette opération peut entraîner une réduction du nombre de lignes (en cas d'absence d'un identifiant de ligne - clé). Dans l'exemple de la table "employé", une projection peut être effectuée en ne retenant que les attributs "nom" et "matricule", soit une table résultante réduite à deux colonnes.

L'opérateur de produit cartésien (binaire) permet de construire une relation dont le schéma est constitué des attributs (libellés des colonnes de la table) des deux relations. Les n-uplets fournis sont construits sur la base du produit cartésien (toutes les combinaisons possibles entre chaque ligne de la première relation et chacune des lignes de la deuxième relation)

À partir de ces 5 opérateurs de base peuvent être définis des opérateurs dérivés (n'apportent aucun pouvoir d'interrogation supplémentaire mais permettent de faciliter les manipulations). L'opérateur d'intersection (binaire) fournit les n-uplets présents dans les deux relations, l'opérateur de jointure permet de construire des séries de tuples comportant un identifiant commun à partir d'un produit cartésien, la division permettant d'exprimer plus facilement un quantificateur universel (une combinaison d'une différence entre relations obtenues en appliquant des produits cartésiens et des intersections).

Le langage support applicatif d'un Système de Gestion de Bases de Données reposant sur un modèle relationnel est SQL.

Règles de gestion d'un système relationnel de base des données

Règle 1

Unicité :

Toute l'information dans la base de données est représentée d'une et une seule manière, à savoir par des valeurs dans des champs de colonnes de tables.

Règle 2

Garantie d'accès :

Toutes les données doivent être accessibles sans ambiguïté. Cette règle est essentiellement un ajustement de la condition fondamentale pour des clefs primaires. Elle indique que chaque valeur scalaire individuelle dans la base de données doit être logiquement accessible en indiquant le nom de la table contenant, le nom de la colonne contenant et la valeur principale primaire de la rangée contenant.

Règle 3

Traitement des valeurs nulles :

Le système de gestion de bases de données doit permettre à chaque champ de demeurer nul (ou vide). Spécifiquement, il doit soutenir une représentation "d'information manquante et d'information inapplicable" qui est systématique, distincte de toutes les valeurs régulières (par exemple, "distincte de zéro ou tous autres nombres," dans le cas des valeurs numériques), et ce indépendamment du type de données. Cela implique également que de telles représentations doivent être gérées par le système de gestion de bases de données d'une manière systématique.

Règle 4

Catalogue lui-même relationnel :

Le système doit supporter un catalogue en ligne, intégré, relationnel, accessible aux utilisateurs autorisés au moyen de leur langage d'interrogation régulier. Les utilisateurs doivent donc pouvoir accéder à la structure de la base de données (catalogue) employant le même langage d'interrogation qu'ils emploient pour accéder aux données de la base de données.

Règle 5

Sous-langage de données :

Le système doit soutenir au moins un langage relationnel qui:

1. a une syntaxe linéaire
2. peut être employé interactivement et dans des programmes d'application,
3. supporte des opérations de définition d'informations supplémentaires (incluant des définitions de vues), de manipulation de données (mise à jour aussi bien que la récupération), de contraintes de sécurité et d'intégrité, et des opérations de gestion de transaction (commencer, valider et annuler une transaction).

Règle 6

Mise à jour des vues :

Toutes les vues pouvant théoriquement être mises à jour doivent pouvoir l'être par le système.

Règle 7

Insertion, mise à jour, et effacement de haut niveau :

Le système doit supporter les opérations par lot d'insertion, de mise à jour et de suppression. Ceci signifie que des données peuvent être extraites d'une base de données relationnelle dans des ensembles constitués par des données issues de plusieurs tuples et/ou de multiples table. Cette règle explique que l'insertion, la mise à jour, et les opérations d'effacement devraient être supportées aussi bien pour des lots de tuples issues de plusieurs tables que juste pour un tuple unique issu d'une table unique.

Règle 8

Indépendance physique :

Les modifications au niveau physique (comment les données sont stockées, si dans les rangées ou les listes liées, etc.) ne nécessitent pas un changement d'une application basée sur les structures.

Règle 9

Indépendance logique :

Les changements au niveau logique (tables, colonnes, rangées, etc) ne doivent pas exiger un changement dans l'application basée sur les structures. L'indépendance de données logiques est plus difficile à atteindre que l'indépendance de données physiques.

Règle 10

Indépendance d'intégrité :

Des contraintes d'intégrité doivent être indiquées séparément des programmes d'application et être stockées dans le catalogue. Il doit être possible de changer de telles contraintes au fur et à mesure sans affecter inutilement les applications existantes.

Règle 11

Indépendance de distribution :

La distribution des parties de la base de données à de diverses localisations doit être invisible aux utilisateurs de la base de données. Les applications existantes doivent continuer à fonctionner avec succès :

1. quand une version distribuée du système de gestion de bases de données est d'abord présentée ; et
2. quand des données existantes sont redistribués dans le système.

Règle 12

Règle de non-subversion :

Si le système fournit une interface avec langage de bas niveau, cette interface ne doit pas permettre de contourner le système (par exemple pour ajouter une contrainte relationnelle de sécurité ou d'intégrité) : ces contraintes doivent être exprimées dans le langage relationnel de haut niveau.

Principe du modèle relationnel

L'idée centrale du modèle relationnel est de décrire un ensemble de données comme une collection de prédicats sur un ensemble fini de variables sous-jacentes, décrivant les contraintes sur les valeurs et les combinaisons de valeurs possibles. Le contenu de l'ensemble de données qui en résulte à un instant t , le modèle conceptuel des données, est transcribable dans une base de données à travers un modèle physique des données. C'est un modèle fini (logique) de la base de données, à savoir un ensemble de relations, une par variable de relation, de telle sorte que tous les prédicats sont satisfaits. Une demande d'informations de la base de données (une requête de base de données) est également un prédicat. Un grand avantage des données construites à partir de ce modèle relationnel est que l'utilisateur peut, dans sa retranscription physique en base de données, y accéder sans savoir où sont physiquement les données ni comment elles sont stockées. C'est un grand avantage par rapport au modèle hiérarchique implémenté dans les bases de données hiérarchiques ou au modèle réseau³.

La modélisation relationnelle et sa transcription en base de données

La modélisation relationnelle une fois achevée permet de matérialiser les relations sous forme de tables à deux dimensions. Chaque colonne possède un identificateur qui représente un domaine. On appelle tuple ou n-uplet un set des valeurs des attributs incoordonnés, c'est-à-dire la ligne de table. La relation peut donc être définie par le set de n-tuples. « Chaque opération relationnelle sur une table génère une nouvelle relation et les opérateurs relationnels – ceux du langage SQL, permettent de décrire le résultat que l'on veut obtenir sans avoir à décrire la procédure nécessaire pour arriver au résultat : on dit que le langage relationnel est « non procédural »¹.

Pour lier les relations entre elles on utilise la notion de clé primaire et de clé étrangère. La clé primaire d'une relation est un attribut ou un ensemble d'attributs qui permet de désigner d'une façon unique un tuple (par exemple l'attribut Référence Client permet d'identifier de façon unique les tuples de la relation Clients). La seule connaissance de la clé primaire permet d'identifier toute ligne dans une table. Par ailleurs, la clé étrangère est un identifiant qui fait référence à une clé unique dans une autre table⁴. Par exemple, dans la Relation Facture, l'attribut Client contient la référence client et donc permet de retrouver toutes les informations du client concerné dans la relation Clients.

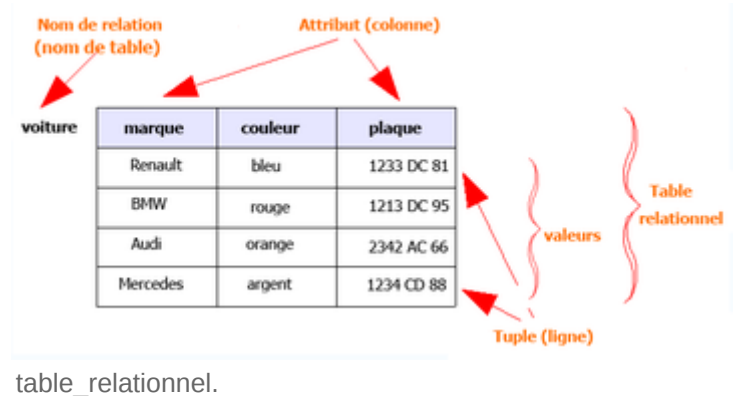
| ID voiture | ID carburant | marque | plaque |
|------------|--------------|---------|------------|
| 1 | 1 | Renault | 1233 DC 81 |
| 2 | 2 | BMW | 1233 DC 95 |
| 2 | 1 | Audi | 2342 AC 66 |

| ID carburant (*) | type |
|------------------|---------|
| 1 | pétrole |
| 2 | gas-oil |
| 3 | gas |

relation.

Au sens mathématique la relation est un sous-ensemble du produit cartésien de certains domaines. Un domaine est représenté comme un ensemble des valeurs : $R = (A1 \times A2 \times A3)$.

Cette relation R est représentée par une table de 3 colonnes (trois attributs) A1, A2, A3 dont chaque ligne est caractérisée par différentes valeurs dans les domaines A1, A2, A3. Pour modéliser l'entité du monde réel « voiture » on prendra comme constituants la marque, la couleur, la plaque d'immatriculation et la date de création : voiture (marque, couleur, plaque-immatriculation, date-création). Les domaines correspondant aux identifiants de colonnes peuvent être déterminés par les ensembles de valeurs suivants :



marque : chaîne de 1 à 50 caractères alphabétiques.

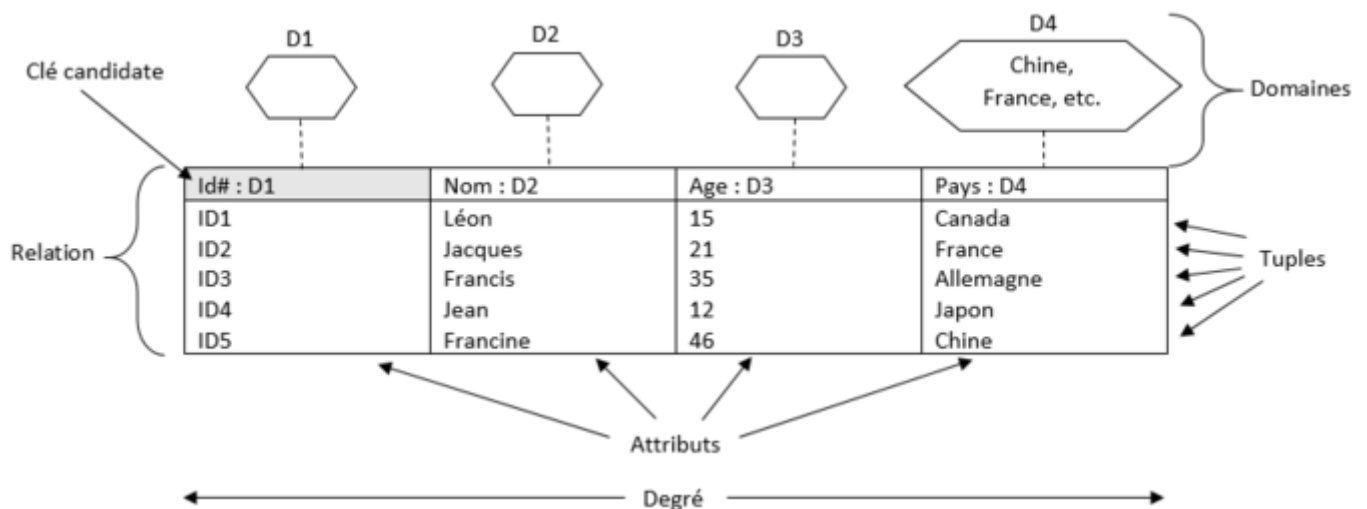
couleur : chaîne de 1 à 30 caractères alphabétiques.

plaque-immatriculation: chaîne de 1 à 10 caractères alphabétiques.

date-création: dates depuis le 1^{er} janvier 1800 jusqu'au présent.

Terminologie structurelle ⁵

Les objets de base souvent référencés dans la modélisation relationnelle sont les domaines, les relations, les attributs, les degrés et les tuples. La figure suivante illustre bien ces concepts :



Domaine

Le domaine représente un ensemble fini de valeur possible pour un attribut donné auquel on définit aussi un ensemble d'opérateurs pouvant être appliqués aux valeurs du domaine. Il faut toutefois faire la distinction entre les domaines et leurs représentations physiques dans le système puisqu'un domaine peut être représenté dans un type de base qui possède des opérations qui ne sont pas représentatives du domaine.

Par exemple, dans la figure ci-dessus, on définit D4 comme l'ensemble des pays de la terre. Alors, puisque l'attribut Pays appartient D4, les valeurs possibles pour l'attribut se limitent aux valeurs définies dans D4 (c.i. : l'ensemble des pays de la terre). Aussi, supposons que le domaine D4 ne définit qu'une seule opération, la mise en majuscule. Dans ce cas, même si le pays est physiquement représenté par une chaîne de caractères il ne sera pas possible d'y appliquer des opérations autres que la mise en majuscule.

Tuple

Grossièrement, un tuple est un enregistrement (une ligne) dans la base de données.

Plus formellement, un tuple est un élément atomique comportant un entête et un corps. L'entête est un ensemble des noms d'attributs et de leurs domaines et le corps est un ensemble de triplets <Nom du domaine, Nom d'attribut, valeur>.

Attributs

Un attribut est simplement la valeur associée à un des triplets d'un tuple.

Clé candidate

Une clé candidate est un ensemble des données permettant d'indexer chaque ligne d'une table donnée de manière différenciée. On utilise souvent une ou plusieurs clés candidates comme clé primaire.

Relation

Une relation (ou table) est un élément constitué d'un entête et d'un corps. L'entête est un ensemble des noms d'attributs et de leurs domaines et le corps est un ensemble de tuples ayant le même entête que la relation.

Attention à ne pas confondre avec le concept de relation entre les tables.

Degré

Le degré est le nombre d'attributs dans une relation.

Cardinalité de relation

Le modèle relationnel prévoit trois types de relations entre tables : 1:1, 1:N et N:N. Les relations entre les tables sont définies dans la colonne partagée. Ce modèle ne soutient pas directement les relations N:N qui seront en fait traduites en deux relations 1:N.

Relation 1:1

Dans deux tables A et B de relation 1:1, un tuple de la table A se rapporte seulement à un tuple de la table B.

Par exemple, un ministre est à la tête d'un ministère et un ministère ne comporte qu'un seul ministre : la table « Ministères » est en relation 1:1 avec la table « Ministres ».

Relation 1:N




Dans deux tables A et B de relation 1:N, un tuple de la table A se rapporte à un ou plusieurs tuples de la table B.

Par exemple, un seul membre de la table « Internats » peut se rapporter à plusieurs membres de la table « Élèves ».

Relation N:N

Dans deux tables A et B de relation N:N, un tuple de la table A se rapporte à un ou plusieurs tuples de la table B et un tuple de la table B se rapporte à un ou plusieurs tuples de la table A. Une relation N:N peut donc être décomposées en deux relations 1:N.

Par exemple, dans une école secondaire, une classe a plusieurs professeurs et un professeur est responsable de plusieurs classes : les tables « Classes » et « Professeurs » sont en relation N:N.

- Sur les autres projets Wikimedia :
-  Modèle relationnel, sur Wikimedia Commons
 -  modèle relationnel, sur le Wiktionnaire
 -  Modèle relationnel, sur Wikiversity